Journal of Applied Logic 12 (2014) 28–44

Contents lists available at ScienceDirect

Journal of Applied Logic

www.elsevier.com/locate/jal

# Modeling Martin-Löf type theory in categories

François Lamarche

*INRIA Saclay – Ile de France, Equipe-Projet Parsifal, France*

ARTICLE INFO

*Article history:*
Available online 4 September 2013

*Keywords:*
Grothendieck fibration
Martin–Löf type theory
Identity type
Path object
Homotopical type theory

ABSTRACT

We present a model of Martin-Löf type theory that includes both dependent products and the identity type. It is based on the category of small categories, with cloven Grothendieck bifibrations used to model dependent types. The identity type is modeled by a path functor that seems to have independent interest from the point of view of homotopy theory. We briefly describe this model's strengths and limitations.



## 1. Introduction

The last few years have seen a flurry of activity in the semantics of Martin-Löf's identity type, based on the fruitful relationship with path objects in homotopy theory.

In this paper we present such a model of identity types in Martin-Löf type theory which has both desirable features of including dependent products and having introduction–elimination operators that are stable under substitution. Moreover our presentation is very concrete, and calculations in the model are fairly easy; in particular no use whatsoever is made of factorization systems, which have been a favored technique in the semantics of identity types [1,19].

This work was first presented at the Makkaifest in Montreal in June 2009. Among the other models that were being developed contemporaneously or semi-contemporaneously, one deserves special mention [5]. Not only does the construction of the simplicial path object described in that paper very much resemble ours—this is not a big surprise, since our model is built on small categories, and a category is a special kind of simplicial set—but also one of its important ingredients is what we have called a triangulator in the present paper. One additional interesting aspect of [5] is that an axiomatic framework is presented for path objects. Our own model almost fits that framework, but not quite, which suggests the existence of a more general framework, that would encompass both approaches. This axiomatic framework also suggests a way to obtain the present model (or something very close to it) by the means of a factorization system.

Independently of its type-theoretical interest, our model seems to have significant interest in the homotopy theory of categories, a subject we intend to investigate.

## 2. What we are looking for

What follows is the structure we will require on a category $\mathscr{C}$ in order to get a model of dependent type theory with an identity predicate. There is nothing original here, except that the presentation is optimized for our purposes.

The first categorical models of dependent types [3] relied on a class of maps $\mathscr{F}$ of $\mathscr{C}$ with the following properties.

- $\mathscr{C}$ has a terminal object,
- all isos in $\mathscr{C}$ are in $\mathscr{F}$,
- $\mathscr{F}$ is closed under pullbacks by arbitrary morphisms of $\mathscr{C}$.

*E-mail address:* francois.lamarche@inria.fr.

1570-8683/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jal.2013.08.003

We will try to use the following notation consistently. We try as much as we can to denote a map $X$ of $\mathscr{F}$ as something like

$$X : \overline{X} \longrightarrow A,$$

where the name of its domain is obtained by overlining the map's name, but the codomain can look arbitrary. Given the above along with an arbitrary $f : B \to A$ the pullback operation is denoted by

$$\begin{array}{ccc} \overline{f^*X} & \longrightarrow & \overline{X} \\ {\scriptstyle f^*X}\downarrow & & \downarrow{\scriptstyle X} \\ B & \xrightarrow[f]{} & A \end{array} \tag{1}$$

The members of $\mathscr{F}$ are most often called *display maps* but we prefer to call them *abstract fibrations*, and we will often just say *fibrations* when the context is clear.

The intuition should be clear and has been used in geometry since the early fifties: given $X$ as above, it is thought of as dependent family $(X_a)_{a\in A}$, and in a concrete category of sets-with-structure the $X_a$ are just the fibers $X^{-1}(a)$. The pullback operation corresponds to substitution: given $f$ as above, then $f^*X$ models the family $(X_{f(b)})_{b\in B}$.

The maps of $\mathscr{F}$ that have the terminal object $\mathbf{1}$ as codomains correspond to ordinary, non-dependent types. In many models all maps to the terminal are in $\mathscr{F}$, but this does not have to be the case in general. Notice that since syntactic entities are built by an inductive process that starts with non-dependent types, the only objects $A$ of $\mathscr{C}$ that appear in the interpretation of a syntactical system are those for which there is a chain $A \to \cdot \to \cdots \to \mathbf{1}$ of display maps to the terminal object. Abstract fibrations whose codomain is a terminal will be just denoted by the source of the domain, since they just correspond to objects of $\mathscr{C}$ and the overline notation becomes cumbersome.

Let $\mathbf{2}$ denote the category with two objects and one arrow between them. Thus $\mathscr{C}^{\mathbf{2}}$ is the familiar category whose objects are maps and whose arrows are commutative squares. It is profitable to think of $\mathscr{F}$ as the full subcategory of $\mathscr{C}^{\mathbf{2}}$, whose objects are the abstract fibrations. The axiom of stability under pullbacks means that the composite

$$\mathscr{F} \longrightarrow \mathscr{C}^{\mathbf{2}} \xrightarrow{\mathrm{Cod}} \mathscr{C} \tag{2}$$

of the inclusion functor with the codomain functor is a "large" Grothendieck fibration.[1] In this context it is natural to call the maps of abstract fibrations that are pullback squares *Cartesian* maps or squares. In the ordinary world of independent type, the categorical version of a unary type constructor is just an endofunction on the class of objects of the modeling category. In the world of dependent types, a unary type constructor is an endofunctor on the category of fibrations and cartesian maps. A type constructor which is a covariant functor is an endofunctor on the category of fibrations and all squares which also sends cartesian maps to cartesian maps. A contravariant type constructor is something a little more elaborate.

To get a completely formalized interpretation of type theory it should be required that the pullback operation be functorial, instead of pseudo-functorial as is the case for ordinary pullbacks in a category. But this requirement is completely independent of the rest and can always be obtained by massaging the target category properly [6], and no further mention of this condition will be made in this paper.

The pair $(\mathscr{C}, \mathscr{F})$ is said to have *dependent products* when the following further condition is obeyed.

- For any $X \in \mathscr{F}$ the pullback functor $X^*$ has a right adjoint, which we denote $\Pi_X$, with the Beck–Chevalley condition holding.

Let us recall the Beck–Chevalley condition: take an arbitrary pullback square of fibrations, as in Eq. (1), denoting the upper horizontal arrow by $h$, and let $Y : \overline{Y} \to \overline{X}$ be another fibration. Beck–Chevalley means that the natural morphism

$$f^*(\Pi_X Y) \to \Pi_{f^*X}\big(h^*Y\big)$$

obtained by

$$\frac{\dfrac{h^*X^*(\Pi_X Y) \xrightarrow{h^*\epsilon} h^*Y}{(f^*X)^*f^*(\Pi_X Y) \longrightarrow h^*Y}}{f^*(\Pi_X Y) \longrightarrow \Pi_{f^*X}(h^*Y)}$$

has to be an isomorphism. This allows us to model the operator $\Pi$ from Martin-Löf type theory.

[1] For the uninitiated reader, the formal definition of Grothendieck fibrations, cartesian maps and cleavages is given at the beginning of Section 3.